Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A process in a data processing system for identifying package names, the process comprising the computer implemented steps of:

responsive to receiving a selection of a class file, identifying [[a]] an absolute path for [[a]] where the class file is actually located;

parsing the absolute path to identify a set of sequential segments; [[and]]

ascertaining a package name for the class file using the set of sequential segments, wherein the package name is ascertained without disassembling the class file; and

loading the class file into a memory of the data processing using the ascertained package name.

2. (Original) The process of claim 1, wherein the class file is on a local file system.

3. (Original) The process of claim 1 further comprising:

receiving a selection of the class file, wherein the selection includes information sufficient to identifying the path for the class file.

4. (Original) The process of claim 1, wherein the parsing step includes:

identifying segments in the set of sequential segments using delimiters in the path.

5. (Currently amended) The process of claim 1, wherein the ascertaining step includes:

selecting first segment containing a base class name to form a proposed package name;

submitting the proposed package name to a [[Java]] virtual machine;

responsive to the proposed package name being an incorrect name, prepending a next segment to the proposed package name; and

responsive to prepending the next segment, submitting the current package name to the [[Java]] virtual machine.

6. (Original) The process of claim 5, wherein the first segment is selected as being a first segment on a right side of the set of sequential segments.

7. (Currently amended) The process of claim 1, wherein the process is located in a [[Java]] class loader.

8. (Currently amended) The process of claim 1, wherein the class file is a [[Java]] class file comprising platform independent bytecodes.

9. (Original) A process in a data processing system for identifying a package name for a class file, the process comprising the computer implemented steps of:

receiving a selection of a class file;

identifying a path for the class file using the selection;

parsing the path to form an ordered set of segments;

selecting an unselected segment from the ordered set of segments;

adding the unselected segment to a set of selected segments;

generating a proposed package name using the set of selected segments;

submitting the proposed package name to a virtual machine for loading; and

repeating the selecting, adding, generating, and submitting steps in response to the proposed package name being an incorrect package name, wherein the package name is identified without examining code in the class file.

10. (Original) The process of claim 9, wherein the code is a set of bytecodes.

11. (Cancelled)

12. (Currently amended) A data processing system for identifying package names, the data processing system comprising:

identifying means responsive to receiving means for receiving a selection of a class file, for identifying [[a]] an absolute path for [[a]] where the class file is actually located;

parsing means for parsing the absolute path to identify a set of sequential segments; and

ascertaining means for ascertaining a package name for the class file using the set of sequential segments, wherein the package name is ascertained without disassembling the class file.

13. (Original) The data processing system of claim 12, wherein the class file is on a local file system.

14. (Original) The data processing system of claim 12 further comprising:

receiving means for receiving a selection of the class file, wherein the selection includes information sufficient to identifying the path for the class file.

15. (Original) The data processing system of claim 12, wherein the identifying means is a first identifying means and wherein the parsing means includes:

second identifying means for identifying segments in the set of sequential segments using delimiters in the path.

16. (Currently amended) The data processing system of claim 12, wherein the ascertaining means includes:

selecting means for selecting first segment containing a base class name to form a proposed package name;

first submitting means for submitting the proposed package name to a [[Java]] virtual machine;

prepending means, responsive to the proposed package name being an incorrect name, for prepending a next segment to the proposed package name; and

second submitting means responsive to prepending the next segment, for submitting the current package name to the [[Java]] virtual machine.

17. (Original) The data processing system of claim 16, wherein the first segment is selected as being a first segment on a right side of the set of sequential segments.

18. (Currently amended) The data processing system of claim 12, wherein the process is located in a [[Java]] class loader.

19. (Currently amended) The data processing system of claim 12, wherein the class file is a [[Java]] class file comprising platform independent bytecodes.

20. (Original) A process in a data processing system for identifying a package name for a class file, the data processing system comprising:

receiving means for receiving a selection of a class file;

identifying means for identifying a path for the class file using the selection;

parsing means for parsing the path to form an ordered set of segments;

selecting means for selecting an unselected segment from the ordered set of segments;

adding means for adding the unselected segment to a set of selected segments;

generating means for generating a proposed package name using the set of selected segments;

submitting means for submitting the proposed package name to a virtual machine for loading; and

repeating means for repeating initiation of the selecting means, adding means, generating means, and submitting means in response to the proposed package name being an incorrect package name, wherein a package name is identified without examining code in the class file.

21. (Original) The data processing system of claim 20, wherein the code is a set of bytecodes.

22. (Cancelled)

23. (Currently amended) A computer program product in a data structure encoded on a computer readable medium and operable with a computing device for identifying package names, the computer program product comprising:

first instructions responsive to receiving a selection of a class file, for identifying [[a]] an absolute path for [[a]] where the class file is actually located;

second instructions for parsing the absolute path to identify a set of sequential segments; [[and]]

third instructions for ascertaining a package name for the class file using the set of sequential segments, wherein the package name is ascertained without disassembling the class file; and

fourth instructions for loading the class file into a memory of the data processing using the ascertained package name.

24. (Original) The computer program product of claim 23, wherein the class file is on a local file system.

25. (Original) The computer program product of claim 23 further comprising:

fourth instructions for receiving a selection of the class file, wherein the selection includes information sufficient to identifying the path for the class file.

26. (Original) The computer program product of claim 23, wherein the second instructions includes:

sub-instructions for identifying segments in the set of sequential segments using delimiters in the path.

27. (Currently amended) The computer program product of claim 23, wherein the third instructions includes:

first sub-instructions for selecting first segment containing a base class name to form a proposed package name;

second sub-instructions for submitting the proposed package name to a [[Java]] virtual machine;

third sub-instructions for responsive to the proposed package name being an incorrect name, prepending a next segment to the proposed package name; and

fourth sub-instructions for responsive prepending the next segment submitting the current package name to the [[Java]] virtual machine.

28. (Original) The computer program product of claim 27, wherein the first segment is selected as being a first segment on a right side of the set of sequential segments.

29. (Currently amended) The computer program product of claim 23, wherein the process is located in a [[Java]] class loader.

30. (Currently amended) The computer program product of claim 23, wherein the class file is a [[Java]] class file comprising platform independent bytecodes.

31. (Currently amended) A computer program product in a data structure encoded on a computer readable medium and operable with a computing device for identifying a package name for a class file, the computer program product comprising:

first instructions for receiving a selection of a class file;

second instructions for identifying a path for the class file using the selection;

third instructions for parsing the path to form an ordered set of segments;

fourth instructions for selecting an unselected segment from the ordered set of segments;

fifth instructions for adding the unselected segment to a set of selected segments;

sixth instructions for generating a proposed package name using the set of selected segments;

seventh instructions for submitting the proposed package name to a virtual machine for loading; and

eighth instructions for repeating the initiation of fourth, fifth, sixth, and seventh instructions in response to the proposed package name being an incorrect package name, wherein a package name is identified without examining code in the class file.

32. (Original) The computer program product of claim 31, wherein the code is a set of bytecodes.

33. (Cancelled)

34. (Currently amended) A data processing system comprising:

a bus system;

a memory connected to the bus system, wherein the memory includes a set of instructions; and

a processing unit connected to the bus system, wherein the processing unit executes a set of instructions to identify [[a]] an absolute path for where a class file is actually located in response to receiving a selection of the class file; to parse the absolute path to identify a set of sequential segments; and to ascertain a package name for the class file using the set of sequential segments, wherein the package name is ascertained without disassembling the class file.

35. (Original) A data processing system comprising:

a bus system;

a memory connected to the bus system, wherein the memory includes a set of instructions; and

a processing unit connected to the bus system, wherein the processing unit executes a set of instructions to receive a selection of a class file; identify a path for the class file using the selection; parse the path to form an ordered set of segments; select an unselected segment from the ordered set of segments; add the unselected segment to a set of selected segments; generate a proposed package name using the set of selected segments; submit the proposed package name to a virtual machine for loading; and repeat instructions to select, add, generate, and submit in response to the proposed package name being an incorrect package name, wherein the package name is identified without examining code in the class file.